SIAM J. COMPUT.
Vol. 9, No. 2, May 1980

© 1980 Society for Industrial and Applied Mathematics
0097-5397/80/0902-0007 $01.00/0

OPTIMAL MERGING OF 3 ELEMENTS WITH n ELEMENTS*

F. K. HWANG†

Abstract. Suppose we are given two disjoint linearly-ordered subsets A_m and B_n of a linearly-ordered set S. The problem is to determine the linear ordering of their union (i.e., to "merge" A_m and B_n) by means of a sequence of pairwise comparisons between an element of A_m and an element of B_n. An algorithm to merge A_m and B_n is called M-optimal if it minimizes the maximum number of comparisons required where the maximum is taken over all possible ordering of $A_m \cup B_n$. Let $f_m(k)$ denote the largest n such that A_m and B_n can be merged in k comparisons by an M-optimal algorithm. The determination of $f_3(k)$ has been an open problem for a long time. In this paper we give a constructive proof that

$$f_3(1)=0,\quad f_3(2)=1,\quad f_3(3)=1,\quad f_3(4)=2,$$
$$f_3(5)=3,\quad f_3(6)=4,\quad f_3(7)=6,\quad f_3(8)=8,$$

and for $r \geqq 3$,

$$f_3(3r)=[\tfrac{43}{7}2^{r-2}]-2,$$
$$f_3(3r+1)=[\tfrac{107}{7}2^{r-3}]-2,$$
$$f_3(3r+2)=\left[\frac{17\cdot 2^r-6}{7}\right]-1.$$

where $[x]$ denotes the integral part of x.

1. Introduction.

Suppose we are given two disjoint linearly-ordered subsets A_m and B_n of a linearly-ordered set S, say

$$A_m=\{a_1<a_2<\cdots<a_m\},$$
$$B_n=\{b_1<b_2<\cdots<b_n\}.$$

The problem is to determine the linear ordering of their union (i.e., to *merge* A_m and B_n) by means of a sequence of pairwise comparisons between an element of A_m and an element of B_n. Given any algorithm g to solve this problem, which we refer to as the (m, n) problem, let $K_g(m, n)$ denote the maximum of comparisons required considering all possible orderings of $A_m \cup B_n$. An algorithm g is said to be *M-optimal* if $K_g(m, n) = K(m, n)$, where $K(m, n) = \min_x K_x(m, n)$ where x ranges over all possible algorithms g.

The determination of $K(m, n)$ for general m and n is still an open problem. The several special classes of the (m, n) problem for which $K(m, n)$ are known are (see [1], [2], [3]):

(i) $m = 1$ and 2.

(ii) $m+4 \geqq n \geqq m$.

In particular, the $(3, n)$ problem has been mentioned as an open problem in [1] and [3]. The purpose of this paper is to settle the $(3, n)$ problem.

Let $f_m(k)$ denote the largest n such that the (m, n) problem can be done in k comparisons. Then for fixed m, $K(m, n)$ is known for every n if and only if $f_m(k)$ is known for every k. It has been determined in the literature [1], [2], [3] that

$$f_1(k)=2^k-1$$

and

$$f_2(k)=\begin{cases}[\frac{12}{7}2^{r-1}]-1 & \text{for } k=2r-1,\\ [\frac{17}{7}2^{r-1}]-1 & \text{for } k=2r.\end{cases}$$

where $[x]$ denotes the largest integer not exceeding x. In this paper, we give the values of

* Received by the editors March 29, 1978, and in revised form January 10, 1979.

† Bell Laboratories, Murray Hill, New Jersey 07974.

$f_3(k)$ as follows:

$$f_3(1)=0,\quad f_3(2)=1,\quad f_3(3)=1,\quad f_3(4)=2,$$
$$f_3(5)=3,\quad f_3(6)=4,\quad f_3(7)=6,\quad f_3(8)=8,$$

and for $r \geqq 3$,

$$f_3(3r)=\left[\tfrac{43}{7}2^{r-2}\right]-2,$$
$$f_3(3r+1)=\left[\tfrac{107}{7}2^{r-3}\right]-2,$$
$$f_3(3r+2)=\left[\frac{17\cdot 2^r-6}{7}\right]-1.$$

A table for $f_2(k)$ and $f_3(k)$ for $k \leqq 18$ is given in the following:

k	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18
$f_2(k)$	0	1	2	3	5	8	12	18	26	37	53	76	108	154	218	309	437	620
$f_3(k)$	0	1	1	2	3	4	6	8	10	13	17	22	28	37	47	59	75	96

2. An algorithm for the $(3, f_3(k))$ problem. At a given stage of the merging process, a certain number of comparisons have been made and partial knowledge on the relations between a_i's and b_j's has been accumulated. We use a diagraph, called a *configuration*, to characterize the known relations at the current stage. In a configuration, each node represents an element and a link $x \to y$ indicates the relation that x is less than y. We also require that the configuration be *minimal* in the sense that no link can be omitted without reducing our knowledge of the relations. A configuration X is said to *dominate* a configuration Y if every relation in X is implied by a relation in Y. Note that X dominates Y implies that the number of comparisons required to do X cannot exceed the number required to do Y.

We present a merging algorithm in the format of a rooted binary tree where each node is associated with a configuration. The root of the tree is associated with the configuration before any comparison is made. Every internal node is also associated with a comparison and its two outlinks represent the two possible outcomes. The two outlinks lead to two nodes each associated with a new configuration absorbing the outcome of the comparison just made. A terminal node is usually associated with a configuration which can be readily analyzed. We indicate the number of comparisons required to merge an a_i (or a group of a_i) under that a_i (or that group) in the configuration and verify it in a later analysis. The sum of these indicated numbers plus the number of comparisons already taken to reach that configuration should not exceed k for an algorithm for the $(3, f_3(k))$ problem.

A few conventions are adopted to facilitate the presentation of a configuration. First of all, we replace the links between elements of $B_n(A_3)$ by undirected edges "—" since these elements are always ordered in a row from left to right and no confusion can arise. Secondly, as n can be very large, to save space, elements of B_n do not appear in the configuration unless they are linked to some element of A_3. Consequently, it is necessary to label the elements of B_n which do appear in the configuration. On the other hand, the elements of A_3 appear in every configuration (ordered in a row), hence need no labeling. Thirdly, the two nodes involved in the current comparison are denoted by solid circles while all other nodes are denoted by empty circles.

The values of $f_3(k)$ for $k \leqq 7$ are known (see [3]). It is also easy to verify that the (3, 8) problem can be done in eight comparisons and the (3, 10) problem in nine. For $k \geqq 10$, we present the algorithm in three parts depending on the residue classes of k mod 3 and using induction on k. We first study the case $k = 3r \geqq 12$. For $r = 4$, the algorithm is illustrated in Fig. 1.

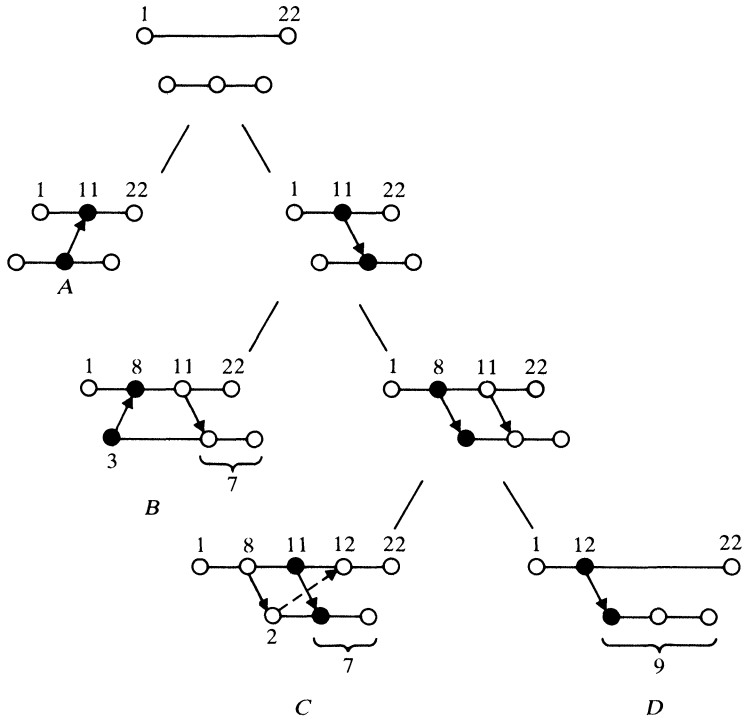


FIG. 1. $k = 12$.

Analysis for the terminal nodes in Fig. 1. A. The merging problem is not affected if the direction of every link is reversed. Therefore, a configuration can be considered either in its own form or the reverse form. In that sense configuration A dominates the configuration to its right.

B, C, D. These are easily checked by using the definitions $f_1(k)$, $f_2(k)$ and $f_3(k)$. For $k = 3r \geqq 15$, the algorithm is given in Fig. 2.

Definitions of symbols used in Fig. 2.

$$c_0 = f_3(3r) = 2^r + f_2(2r-3) + f_2(2r-7) + 1 = [\tfrac{43}{7}2^{r-2}] - 2,$$

$$c_1 = \left[\frac{f_3(3r)+1}{2}\right] = \left[\frac{43 \cdot 2^{r-3} - 4}{7}\right],$$

$$c_2 = 2^{r-1} + 2^{r-2} + f_3(3r-4) + 1 = \left[\frac{38 \cdot 2^{r-2} - 6}{7}\right],$$

$$c_3 = f_3(3r) - 2^{r-4} + 1 = [\tfrac{165}{7}2^{r-4}],$$

$$c_4 = 2^{r-1} + 2^{r-2} + f_2(2r-3) + 1 = [\tfrac{33}{7}2^{r-2}],$$

$$c_5 = c_4 + 2^{r-3} = [\tfrac{73}{7}2^{r-3}],$$

$$c_6 = c_5 + 2^{r-4} = [\tfrac{153}{7}2^{r-4}],$$

$$c_3' = c_5' = 2^{r-1},$$

$$c_4' = c_6' = 2^{r-1} + 2^{r-2} = 3 \cdot 2^{r-2}.$$

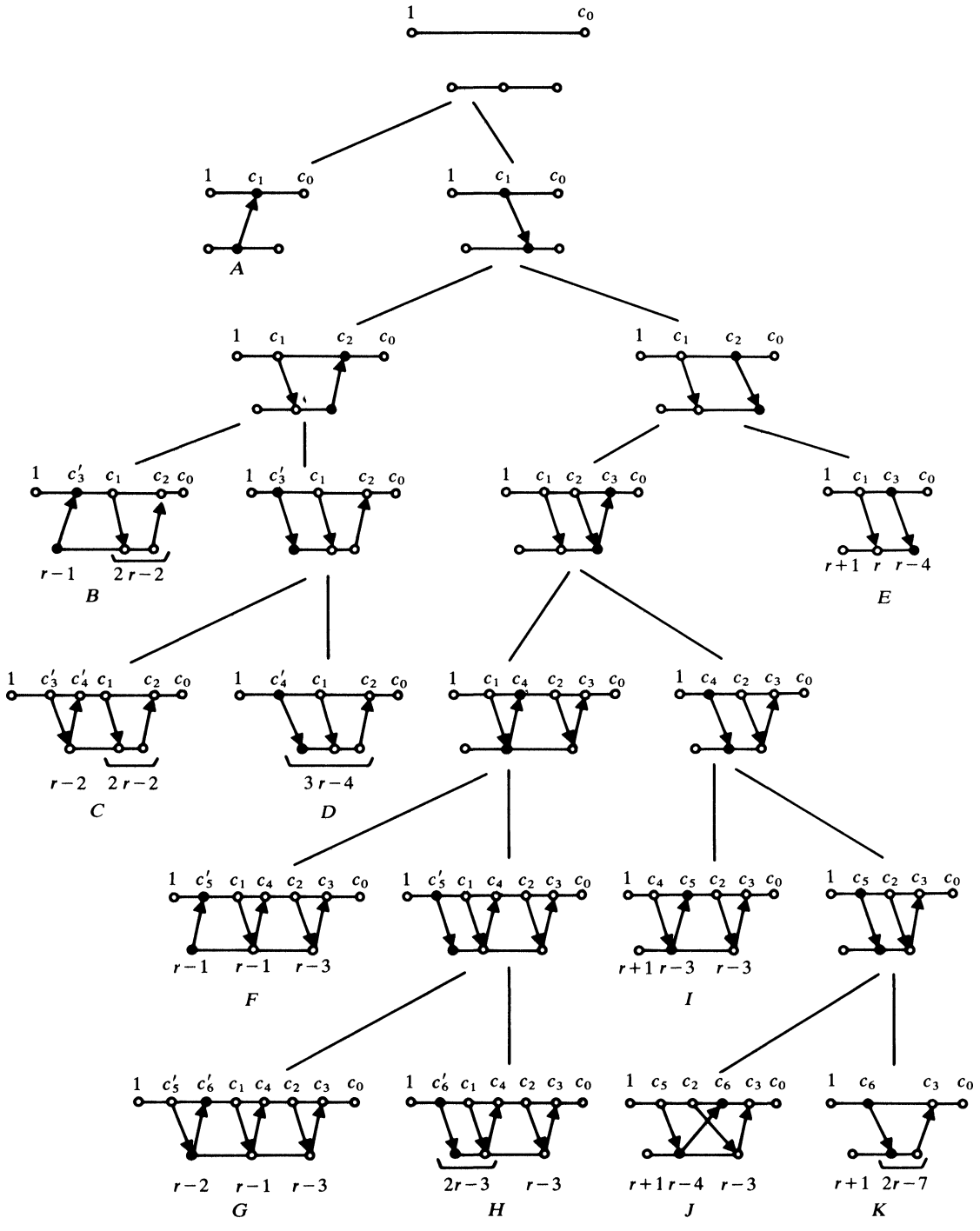


FIG. 2. $k = 3r \geqq 15$.

It is clear that

$$c_0 \geqq c_3 \geqq c_6 \geqq c_2 \geqq c_5 \geqq c_4 \geqq c_1 \geqq c_4' \geqq c_3'.$$

Analysis for the terminal nodes in Fig. 2.

A. Configuration A dominates the configuration to its right.

B. a_1: $c_3' - 1 = f_1(r-1)$. a_2 and a_3: $c_2 - c_1 - 1 \leqq [\frac{33}{7}2^{r-3}] \leqq [\frac{17}{7}2^{r-2}] - 1 = f_2(2r-2)$.

C. a_1: $c_4' - c_3' - 1 = f_1(r-2)$.

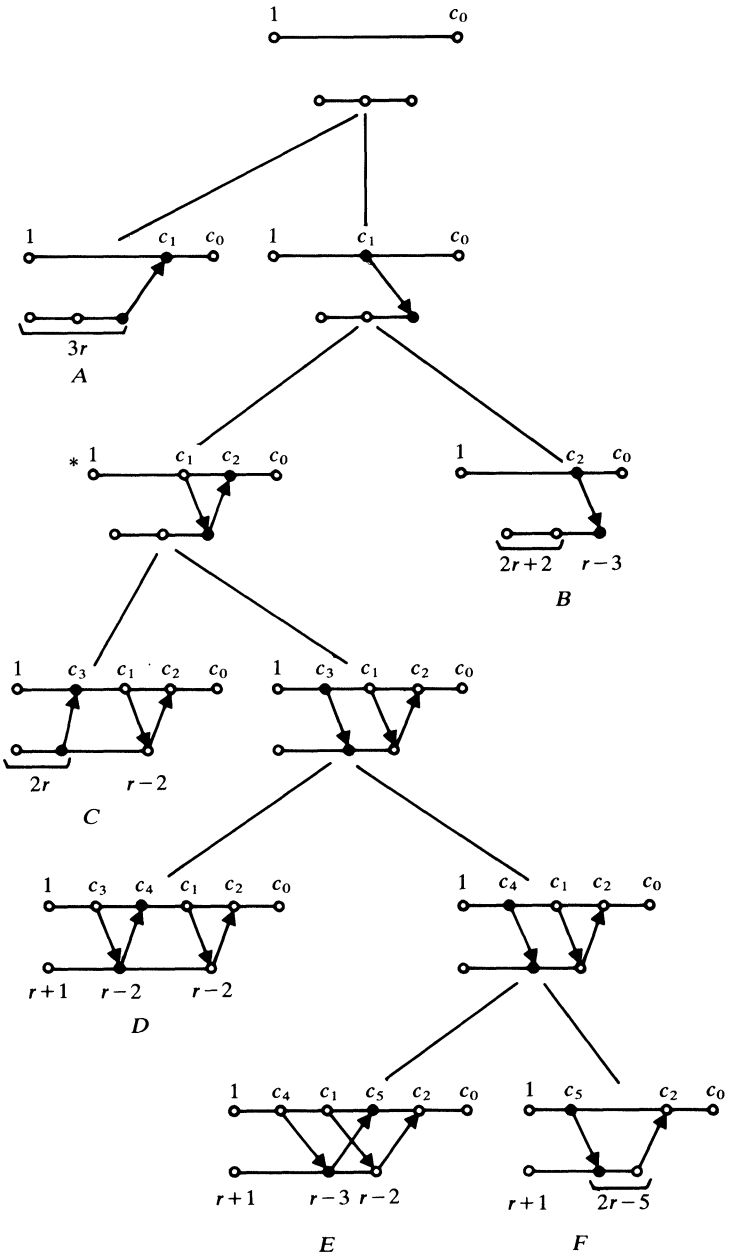


FIG. 3. $k = 3r+1 \geqq 10$.

D. Configuration D certainly dominates the configuration D′ obtained from D by deleting the link $c_1 \to a_2$. Since $c_2 - c_4' - 1 = f_3(3r-4)$, D′, hence D, can be done in $3r-4$ comparisons.

E. a_3: $c_0 - c_3 = f_1(r-4)$. a_2: $c_0 - c_1 \leqq [(43 \cdot 2^{r-3} - 4)/7] \leqq 2^r - 1 = f_1(r)$. a_1: $c_0 = [\frac{43}{7}2^{r-2}] - 2 \leqq 2^{r+1} - 1 = f_1(r+1)$.

F. a_3: $c_3 - c_2 - 1 \leqq [\frac{13}{7}2^{r-4}] \leqq 2^{r-3} - 1 = f_1(r-3)$. a_2: $c_4 - c_1 \leqq [\frac{23}{7}2^{r-3}] \leqq 2^{r-1} - 1 = f_1(r-1)$. a_1: $c_5' - 1 = f_1(r-1)$.

G. This is similar to F.

H. Ignoring the link $c_1 \to a_2$, then since $c_4 - c_6' - 1 = f_2(2r-3)$, a_1 and a_2 together can be merged in $2r-3$ comparisons.

I. a_2: $c_5 - c_4 - 1 = f_1(r-3)$.

J. a_2: $c_6 - c_5 - 1 = f_1(r-4)$.

K. a_2 and a_3: $c_3 - c_6 - 1 = f_3(3r) - 2^{r-4} - (2^{r-1} + 2^{r-2} + f_2(2r-3) + 1 + 2^{r-3} + 2^{r-4}) = f_2(2r-7)$.

Next we study the case $k = 3r+1 \geqq 10$. The algorithm is given in Fig. 3.

Definitions of symbols used in Fig. 3.

$$c_0 = f_3(3r+1) = 2^{r-2} + 2^{r-3} + f_3(3r) = [\tfrac{107}{7}2^{r-3}] - 2,$$

$$c_1 = f_3(3r) + 1 = [\tfrac{43}{7}2^{r-2}] - 1,$$

$$c_2 = c_1 + 2^{r-2} = [\tfrac{50}{7}2^{r-2}] - 1,$$

$$c_3 = f_2(2r) + 1 = [\tfrac{17}{7}2^{r-1}],$$

$$c_4 = c_3 + 2^{r-2} = [\tfrac{41}{7}2^{r-2}],$$

$$c_5 = c_4 + 2^{r-3} = [\tfrac{89}{7}2^{r-3}].$$

It is clear that

$$c_0 \geqq c_2 \geqq c_5 \geqq c_1 \geqq c_4 \geqq c_3.$$

Analysis for the terminal nodes in Fig. 3.

A. a_1, a_2 and a_3: $c_1 - 1 = f_3(3r)$.

B. a_3: $c_0 - c_2 = f_1(r-3)$. a_1 and a_2: $c_0 = [\frac{107}{7}2^{r-3}] - 2 \leqq [\frac{17}{7}2^r] - 1 = f_2(2r+2)$.

C. a_3: $c_2 - c_1 - 1 = f_1(r-2)$. a_1 and a_2: $c_3 - 1 = f_2(2r)$.

D. a_2: $c_4 - c_3 - 1 = f_1(r-2)$. a_1: $c_4 - 1 = [\frac{41}{7}2^{r-2}] - 1 \leqq 2^{r+1} - 1 = f_1(r+1)$.

E. a_2: $c_5 - c_4 - 1 = f_1(r-3)$. a_1: $c_5 - 1 = [\frac{89}{7}2^{r-3}] - 1 \leqq 2^{r+1} - 1 = f_1(r+1)$.

F. a_2 and a_3: $c_2 - c_5 - 1 \leqq [\frac{11}{7}2^{r-3}] - 1 \leqq [\frac{12}{7}2^{r-3}] - 1 = f_2(2r-5)$. a_1: $c_2 - 1 = [\frac{50}{7}2^{r-2}] - 2 \leqq 2^{r+1} - 1 = f_1(r+1)$.

Note that comparison 3, or comparisons 3 and 4 together, can be taken before comparison 2 without affecting the results.

Finally, we study the case $k = 3r+2 \geqq 11$. The algorithm is given in Fig. 4.

Definitions of symbols used in Fig. 4.

$$c_0 = f_3(3r+2) = 2f_2(2r) + 1 = \left[\frac{17 \cdot 2^r - 6}{7}\right] - 1,$$

$$c_1 = \left[\frac{c_0 + 1}{2}\right] = \left[\frac{17}{7}2^{r-1}\right],$$

$$c_2 = 2^r,$$

$$c_3 = c_2 + 2^{r-1} = 3 \cdot 2^{r-1}.$$

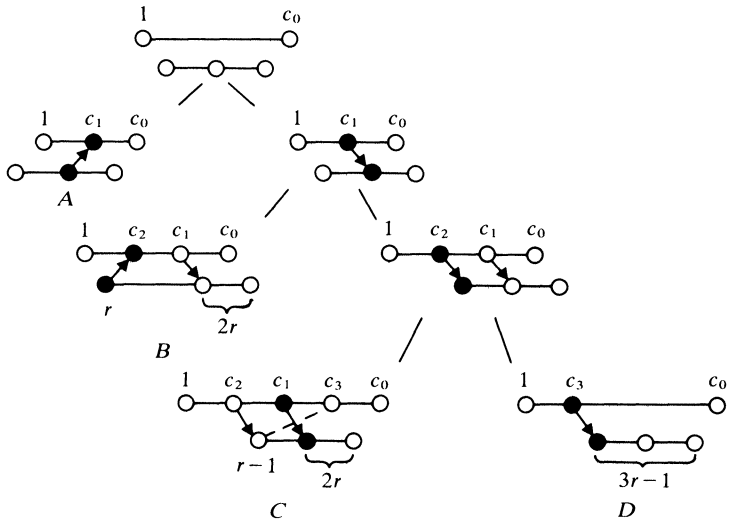


FIG. 4. $k = 3r+2 \geqq 11$.

Analysis for the terminal nodes in Fig. 4.

A. Configuration A is symmetric to the configuration to its right.

B. a_1: $c_2 - 1 = 2^r - 1 = f_1(r)$. a_2 and a_3: $c_0 - c_1 = f_2(2r)$.

C. a_1: $c_3 - c_2 - 1 = 2^{r-1} - 1 = f_1(r-1)$.

D. a_1, a_2 and a_3: $c_0 - c_3 = [(13 \cdot 2^{r-1} - 6)/7] - 1 \leqq [(17 \cdot 2^{r-1} - 6)/7] - 1 = f_3(3r-1)$.

3. Optimality of the algorithm. To prove the optimality of the algorithm, we have to show that given a $(3, f_3(k)+1)$ problem, then regardless of which comparison a_i vs. b_j we start with, the problem cannot be done in k comparisons. Since there are too many choices of a_i vs. b_j to be dealt with, we use the following scheme to simplify things. We first classify all the possible choices into three categories depending on which a_i, $i = 1, 2, 3$, the comparison involves. Then we deal with each category by using the following lemma.

LEMMA 0. *Suppose we can show for fixed a_i, neither of the following two cases can be done in k comparisons*: (i) *The first comparison is a_i vs. b_x and results in $a_i \to b_x$.*

(ii) *The first comparison is a_i vs. b_{x-1} and results in $a_i \leftarrow b_{x-1}$.*

Then any algorithm whose first comparison involves a_i cannot be done in k comparisons.

Proof. Suppose an algorithm starts with the first comparison a_i vs. b_y. If $y \geqq x$, consider the outcome that $a_i \to b_y$. Then the resultant configuration is dominated by the configuration stated in (i) of Lemma 0. If $y \leqq x-1$, consider the outcome that $a_i \leftarrow b_y$. Then the resultant configuration is dominated by the configuration stated in (ii) of Lemma 0. In either case, at least one outcome will result in more than k comparisons, hence the Lemma.

The proof of the optimality of the algorithm will again be illustrated as a rooted tree (but no longer a binary tree). There will be two types of nodes in the tree: a *configuration node* shows the configuration after the current comparison, and an *index node* indexes the history of comparisons before a configuration node is reached. The index used is the ordered sequence of a_i's the comparisons have been involved including the current one. We also indicate the outcomes of the previous comparisons by using $\bar{a}_i$ ($\underline{a}_i$) to denote the outcome that $a_i \leftarrow b_x$ ($a_i \to b_x$) for some b_x. Thus the tree starts with a configuration node showing the configuration before any comparison is made. Then it branches into three

index nodes labeled by a_1, a_2 and a_3 respectively. Each index node, say, the one labeled by a_i, now branches into two configuration nodes showing the configurations obtained after the comparison $a_i \leftarrow b_x$ and the comparison $a_i \rightarrow b_{x-1}$ respectively. Each configuration node then branches into three more index nodes and the branching process goes on. If the configuration associated with a configuration node can be readily analyzed, that node becomes a terminal node with no more branching from it. Again we indicate the least numbers of comparisons needed to merge such a_i or a group of a_i's under the a_i's. The sum of these numbers plus the number of comparisons already taken to reach that configuration should exceed k in our proof. The following lemmas are crucial for our analysis.

LEMMA 1. *Suppose that a configuration contains the two links* $a_i \rightarrow b_x$ *and* $a_{i+1} \leftarrow b_y$ *for some* i *and* $y \geqq x-1$. *Then the configuration can be replaced by two subconfigurations, one induced by the nodes* $(a_1, \cdots, a_i)$ *and* $(b_1, b_2, \cdots, b_{x-1})$, *and the other by the nodes* $(a_i, \cdots, a_3)$ *and* $(b_y, b_{y+1}, \cdots)$.

Proof. The decomposition is possible since the two subproblems are completely independent.

Sometimes Lemma 1 does not apply directly to a configuration unless an additional link is added. We use a broken link to indicate that the configuration we wish to consider is the one when the broken link is added (and any link now carrying redundant information should be removed). Since it is obvious that the new configuration dominates the original one, it suffices to show that the new configuration cannot be done in a designated number of comparisons.

For many terminal nodes, an application of Lemma 1 to the configurations in there splits the configurations into several subconfigurations each of which can be considered as the starting configuration of a $(1, t)$ problem or a $(2, t)$ problem for some number t. Therefore we need only use the definitions of $f_1(t)$ and $f_2(t)$ to determine how many comparisons are needed for these subconfigurations. As t, $f_1(t)$ and $f_2(t)$ are all explicit, checking is straightforward and can be verified by the reader. We will give analysis only for those subconfigurations whose numbers of comparisons cannot be determined by just knowing $f_1(t)$ and $f_2(t)$. To be more specific, if $N(x, y)$ denotes the number of comparisons required to do the configuration of the following type, then our analysis usually is to determine $N(x, y)$ for some given x and y. The following two lemmas play a crucial role in that kind of analysis.

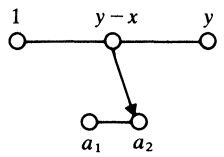


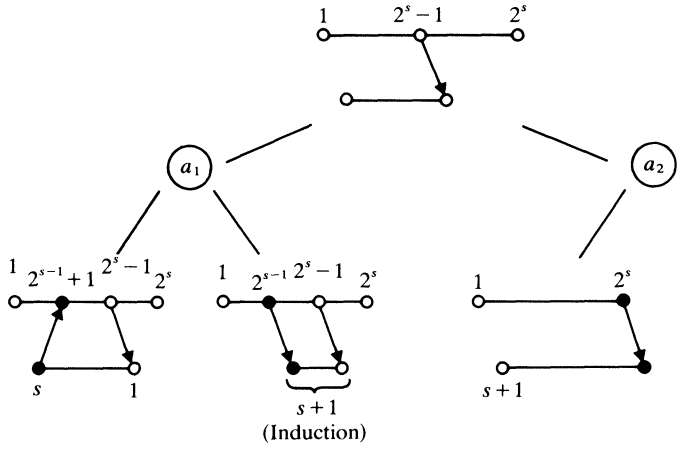
FIG. 5. *An* $N(x, y)$ *configuration.*

LEMMA 2. $N(2^t, 2^s - 1 + 2^t) > s + t + 1$ *for* $s, t \geqq 0$.

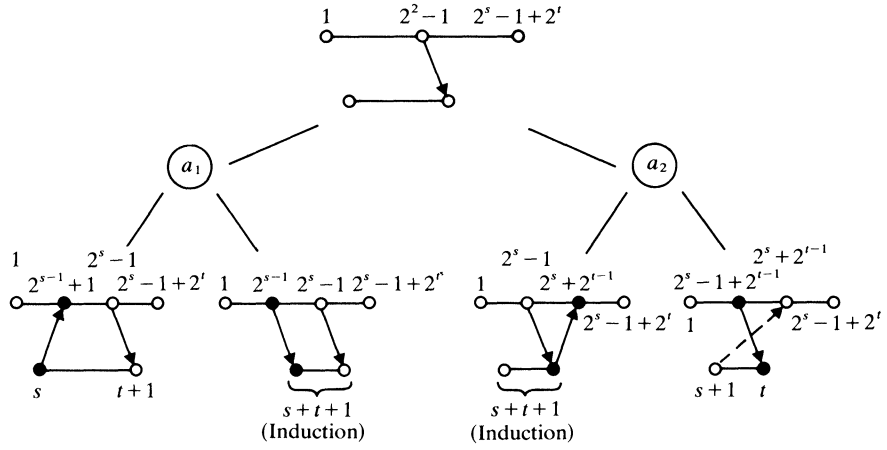
Proof. Suppose $s = 0$. Then $N(2^t, 2^t)$ is the number of comparisons required for the $(2, 2^t)$ problem. Since $f_2(t+1) < 2^t$, Lemma 2 follows immediately. Therefore we assume $s > 0$. The proof of Lemma 2 is given in Fig. 6(a) and 6(b) for the two cases $t = 0$ and $t > 0$ respectively.

LEMMA 3. *For* $k \geqq t + q + 1$ *and* $t - 2 \geqq q \geqq 0$, *define*

$$g(k, t, q) = 2^{t-2} + 2^{t-3} + \cdots + 2^{q+1} + 2^{k-t-1} + 2^{k-t-2} + \cdots + 2^{q+1} + f_2(2q+3) + 1.$$



(a) $t=0$



(b) $t>0$

FIG. 6. *Proof for Lemma* 2.

Then

$$N(x, y)>k \quad \text{if} \quad x\geqq 2^{t-1}+2^q \quad \text{and} \quad y\geqq g(k, t, q).$$

Proof. Since the configuration corresponding to $N(2^{t-1}+2^q, g(k, t, q))$ dominates the configuration corresponding to $N(x, y)$, it suffices to prove

$$N(2^{t-1}+2^q, g(k, t, q))>k.$$

The proof is given by the tree in Fig. 7.

Definitions of symbols used in Fig. 7

$$c_0=g(k, t, q),$$
$$c_1=g(k, t, q)-2^{t-1}-2^q,$$
$$c_2=2^{k-t-1}, \qquad c_3=c_0-2^{t-2}.$$

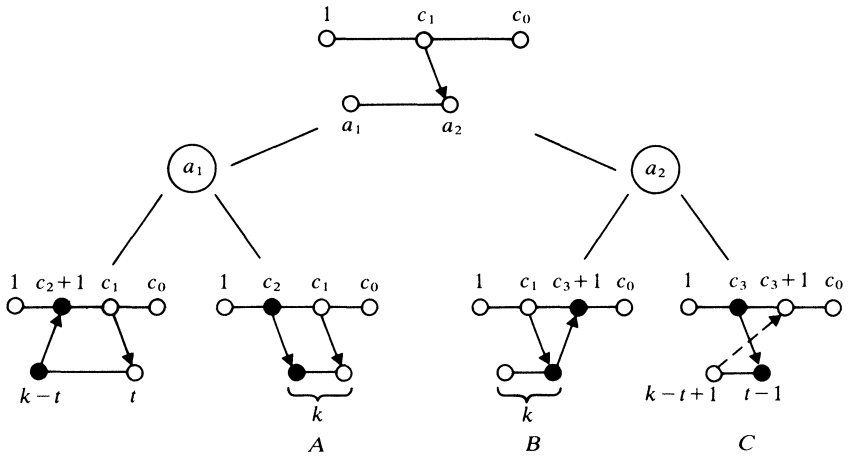


FIG. 7. *Proof for Lemma* 3.

Analysis for the labeled terminal nodes:

A. Since

$$c_0 - c_2 = g(k-1, t, q)$$

and

$$c_0 - c_1 = 2^{t-1} + 2^q.$$

Configuration A needs at least k comparisons by induction.

B. For $q < t-2$,

$$c_3 = g(k, t, q) - 2^{t-2} = g(k-1, t-1, q)$$

and

$$c_3 - c_1 = 2^{t-2} + 2^q.$$

Configuration B needs at least k comparisons by induction.

For $q = t-2$, consider the outcome $a_1 < c_1 + 1$. Then a_2 needs t comparisons and a_1 needs $k-t$ comparisons to merge by using Lemma 1 and the fact

$$\begin{aligned} c_1 &= 2^{k-t-1} + 2^{k-t-2} + \cdots + 2^{t-1} + f_2(2t-1) + 1 - 2^{t-1} - 2^{t-2} \\ &= 2^{k-t-1} + 2^{k-t-2} + \cdots + 2^{t-1} + [\tfrac{3}{7}2^{t-2}] \geqq f_1(k-t-1) + 1. \end{aligned}$$

C. For $q \leqq k-t-2$,

$$\begin{aligned} c_3 &= g(k, t, q) - 2^{t-2} \\ &\geqq 2^{k-t-1} + 2^{k-t-2} + \cdots + 2^{q+1} + f_2(2q+3) + 1 - 2^q \\ &= 2^{k-t} + [\tfrac{3}{7}2^q] \geqq f_1(k-t) + 1. \end{aligned}$$

For $q \geqq k-t-1$,

$$\begin{aligned} c_3 &= g(k, t, q) - 2^{t-2} \\ &\geqq f_2(2q+3) + 1 - 2^q \\ &= [\tfrac{12}{7}2^{q+1}] - 2^q \geqq 2^{q+1} \geqq f_1(k-t) + 1. \end{aligned}$$

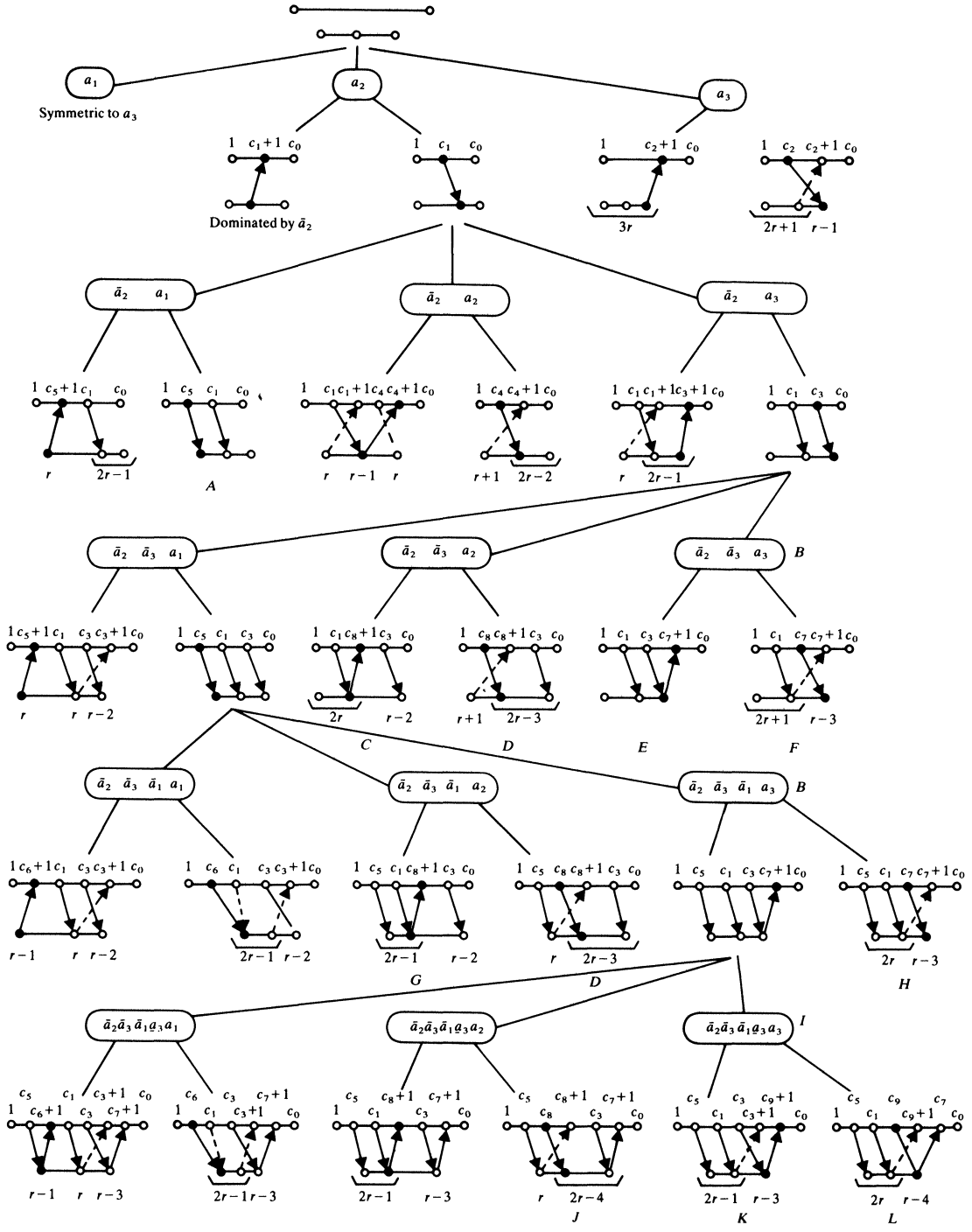


FIG. 8. *Proof for* $k = 3r \geqq 9$.

We are now ready to prove the optimality of the algorithm. The optimality of the algorithm for $k \leqq 7$ is well documented in [3]. Therefore, we will only be concerned with the case $k \geqq 8$. Again, the proof is separated into three parts depending on the residue classes of k mod 3. We also define $f_1(r) = -1$ for negative r to avoid separate discussions for certain special cases. We first study the case $K = 3r \geqq 9$. The proof is illustrated in Fig. 8.

Definitions of symbols used in Fig. 8.

$$c_0 = f_3(3r) + 1 = [\tfrac{43}{7}2^{r-2}] - 1,$$

$$c_1 = \left\lceil \frac{c_0 - 1}{2} \right\rceil = [\tfrac{43}{7}2^{r-3}],$$

$$c_2 = c_0 - 2^{r-2} = [\tfrac{36}{7}2^{r-2}] - 1,$$

$$c_3 = c_1 + f_2(2r-2) + 1 = 11 \cdot 2^{r-3} - 1,$$

$$c_4 = c_1 + 2^{r-2} = [\tfrac{57}{7}2^{r-3}],$$

$$c_5 = 2^{r-1},$$

$$c_6 = c_5 + 2^{r-2} = 3 \cdot 2^{r-2},$$

$$c_7 = c_0 - f_1(r-4) - 1 = \begin{cases} c_0 & \text{for } r = 3, \\ [\tfrac{165}{7}2^{r-4}] - 1 & \text{for } r \geqq 4, \end{cases}$$

$$c_8 = 2^{r-1} + 2^{r-2} + f_2(2r-3) + 1 = [\tfrac{33}{7}2^{r-2}],$$

$$c_9 = c_7 - f_1(r-5) - 1 = \begin{cases} c_7 & \text{for } r = 3, 4, \\ [\tfrac{323}{7}2^{r-5}] - 1 & \text{for } r \geqq 5. \end{cases}$$

Analysis for the labeled nodes in Fig. 8. (It is understood that Lemma 3 is to be applied in each case and we just verify the conditions of Lemma 3 in the following.)

A. If the next comparison involves a_3, future comparisons are same as (except a comparison of a_1 with c_5 or c_{5+1} should be ignored) and analysis is analogous to the $\bar{a}_2\bar{a}_3$ case. If the next comparison involves a_2, comparison is same as and analysis is analogous to the $\bar{a}_2 a_2$ case. If the next comparison involves a_1, we consider the two cases $a_1 < c_6 + 1$ and $a_1 > c_6$. In the former case, a_1 needs $r-1$ more comparisons while a_2 and a_3 together needs $2r-1$ comparisons. In the latter case, assume $a_1 > c_1$. Then the problem is reduced to a $(3, c_0 - c_1) = (3, [\tfrac{43}{7}2^{r-3}] - 1)$ which needs $3r-2$ more comparisons since $f_3(3r-3) = [\tfrac{43}{7}2^{r-3}] - 2$ by induction.

B. For $r = 3$, $c_9 = c_0$. Therefore the cases $\bar{a}_2\bar{a}_3\underline{a}_3$ and $\bar{a}_2\bar{a}_3\bar{a}_1\underline{a}_3$ cannot occur.

C. $c_8 - c_1 = [\tfrac{23}{7}2^{r-3}] \geqq 2^{r-2} + 2^{r-3}$, $c_8 = [\tfrac{33}{7}2^{r-2}] = g(2r-1, r-1, r-3)$.

D. Trivially true for $r = 3, 4$. For $r \geqq 5$, $c_0 - c_3 = [\tfrac{9}{7}2^{r-3}] \geqq 2^{r-3} + 2^{r-5}$, $c_0 - c_8 = [\tfrac{10}{7}2^{r-2}] = g(2r-4, r-2, r-5)$.

E. If the next comparison involves a_3, consider the two cases $a_3 < c_9 + 1$ and $a_3 > c_9$. Analysis is analogous to J and K. If the next comparison involves a_2, comparison is same as and analysis is analogous to the case $\bar{a}_2\bar{a}_3 a_2$. If the next comparison involves a_1, future comparisons are same as (except a comparison of a_3 with c_7 or $c_7 + 1$ should be ignored) and analysis is analogous to the $\bar{a}_2\bar{a}_3 a_1$ case.

F. $c_7 - c_1 \geqq 2^{r-1} + 2^{r-3}$, $c_7 \geqq [\tfrac{5}{7}2^{r+1}] = g(2r, r, r-3)$.

G. $c_8 - c_1 = [\tfrac{23}{7}2^{r-3}] \geqq 2^{r-2} + 2^{r-3}$, $c_8 - c_5 = c_8 - 2^{r-1} = g(2r-1, r-1, r-3) - 2^{r-1} = g(2r-2, r-1, r-3)$.

H. Since $c_7 \geqq c_9$, H needs at least as many comparisons as L.

I. For $r=4$, $c_9 = c_0$. Therefore $\bar{a}_2\bar{a}_3\bar{a}_1\underline{a}_3\underline{a}_3$ cannot occur.

J. $c_7 - c_3 \geqq 2^{r-4}+2^{r-5}$, $\quad c_7 - c_8 = c_0 - c_8 - 2^{r-4} = g(2r-4, r-2, r-5) - 2^{r-4} = g(2r-5, r-3, r-5)$.

K. Since $c_3 \geqq c_8$, K needs at least as many comparisons as G.

L. For $r \geqq 4$, $c_9 - c_1 \geqq 2^{r-1}+2^{r-3}$, $c_9 \geqq [\frac{5}{7}2^{r+1}] = g(2r, r, r-3)$.

Next we study the case $k = 3r+1 \geqq 10$. The proof is illustrated in Fig. 9.

Definitions of symbols used in Fig. 9.

$$c_0 = f_3(3r+1)+1 = [\tfrac{107}{7}2^{r-3}]-1,$$

$$c_1 = c_0 - 2^{r-3} = [\tfrac{25}{7}2^{r-1}]-1,$$

$$c_2 = c_1 - f_1(r-4) - 1 = \begin{cases} c_1 & \text{for } r=3, \\ [\tfrac{193}{7}2^{r-4}]-1 & \text{for } r \geqq 4, \end{cases}$$

$$c_3 = c_1 - 2^{r-3} = [\tfrac{93}{7}2^{r-3}]-1,$$

$$c_4 = c_5 - f_1(r-4) - 1 = \begin{cases} c_5 & \text{for } r=3, \\ [\tfrac{179}{7}2^{r-4}]-1 & \text{for } r \geqq 4, \end{cases}$$

$$c_5 = c_3 - 2^{r-3} = [\tfrac{43}{7}2^{r-2}]-1,$$

$$d_1 = 2^{r-1},$$

$$d_2 = d_1 + 2^{r-3} = 5 \cdot 2^{r-3},$$

$$d_3 = d_2 + f_1(r-4) + 1 = \begin{cases} d_2 & \text{for } r=3, \\ 11 \cdot 2^{r-4} & \text{for } r \geqq 4, \end{cases}$$

$$d_4 = d_3 + f_1(r-5) + 1 = \begin{cases} d_3 & \text{for } r=3, 4, \\ 23 \cdot 2^{r-5} & \text{for } r \geqq 5, \end{cases}$$

$$d_5 = d_1 + f_2(2r-4) + 1 = [\tfrac{45}{7}2^{r-3}],$$

$$d_6 = d_1 + 2^{r-1} = 2^r,$$

$$d_7 = d_6 + 2^{r-2} = 5 \cdot 2^{r-2},$$

$$d_8 = d_7 + 2^{r-3} = 11 \cdot 2^{r-3},$$

$$d_9 = d_8 + f_1(r-4) + 1 = \begin{cases} d_8 & \text{for } r=3, \\ 23 \cdot 2^{r-4} & \text{for } r \geqq 4, \end{cases}$$

$$e_1 = 2^r (= d_6),$$

$$e_2 = f_2(2r) + 1 = [\tfrac{17}{7}2^{r-1}],$$

$$e_3 = c_0 - 2^{r-2} - 2^{r-3} - f_2(2r-4) - 1 = [\tfrac{69}{7}2^{r-3}],$$

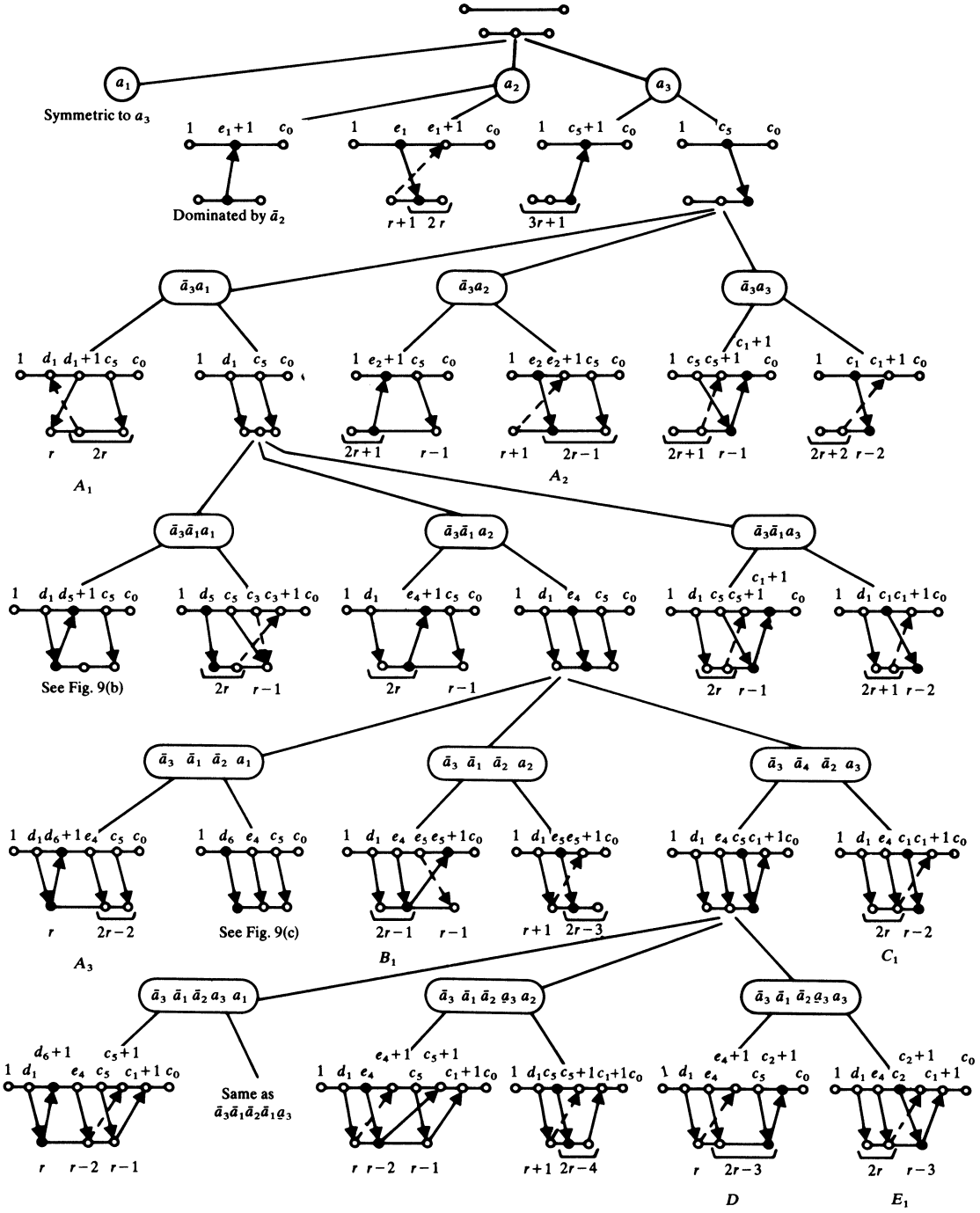
$$e_4 = d_1 + f_2(2r-1) = [\tfrac{19}{7}2^{r-1}],$$

$$e_5 = c_1 - f_2(2r-5) - 1 = [\tfrac{25}{7}2^{r-1}] - [\tfrac{3}{7}2^{r-1}] - 1,$$

$$e_6 = c_0 - f_2(2r-4) - 1 = [\tfrac{45}{7}2^{r-2}].$$

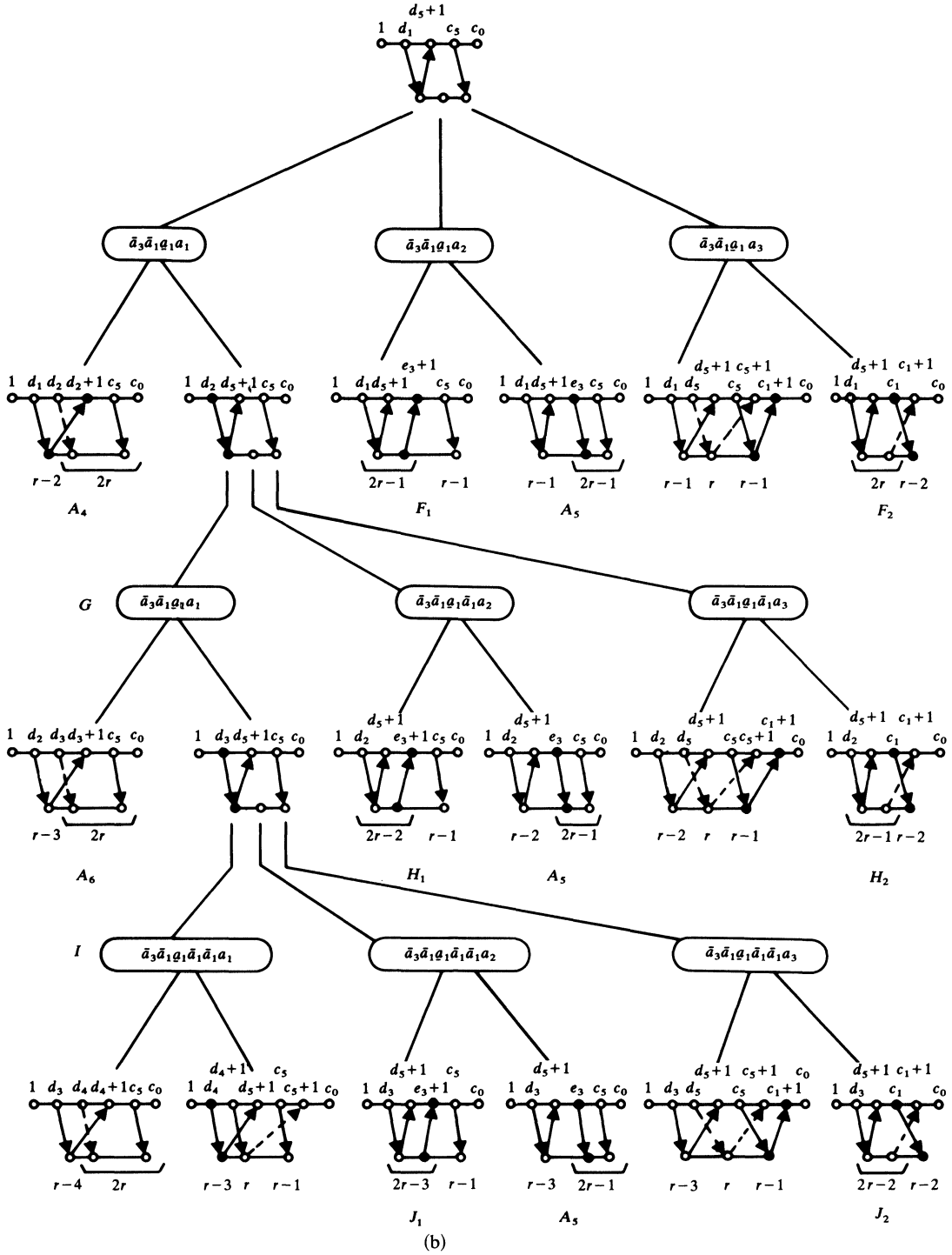
It is clear that

$$c_0 \geqq c_1 \geqq c_2 \geqq c_3 \geqq e_6 \geqq c_4 \geqq e_5 \geqq c_5 \geqq d_9 \geqq e_4 \geqq e_3 \geqq e_2 \geqq d_8 \geqq d_7 \geqq d_6 = e_1 \geqq d_5 \geqq d_4 \geqq d_3 \geqq d_2 \geqq d_1.$$



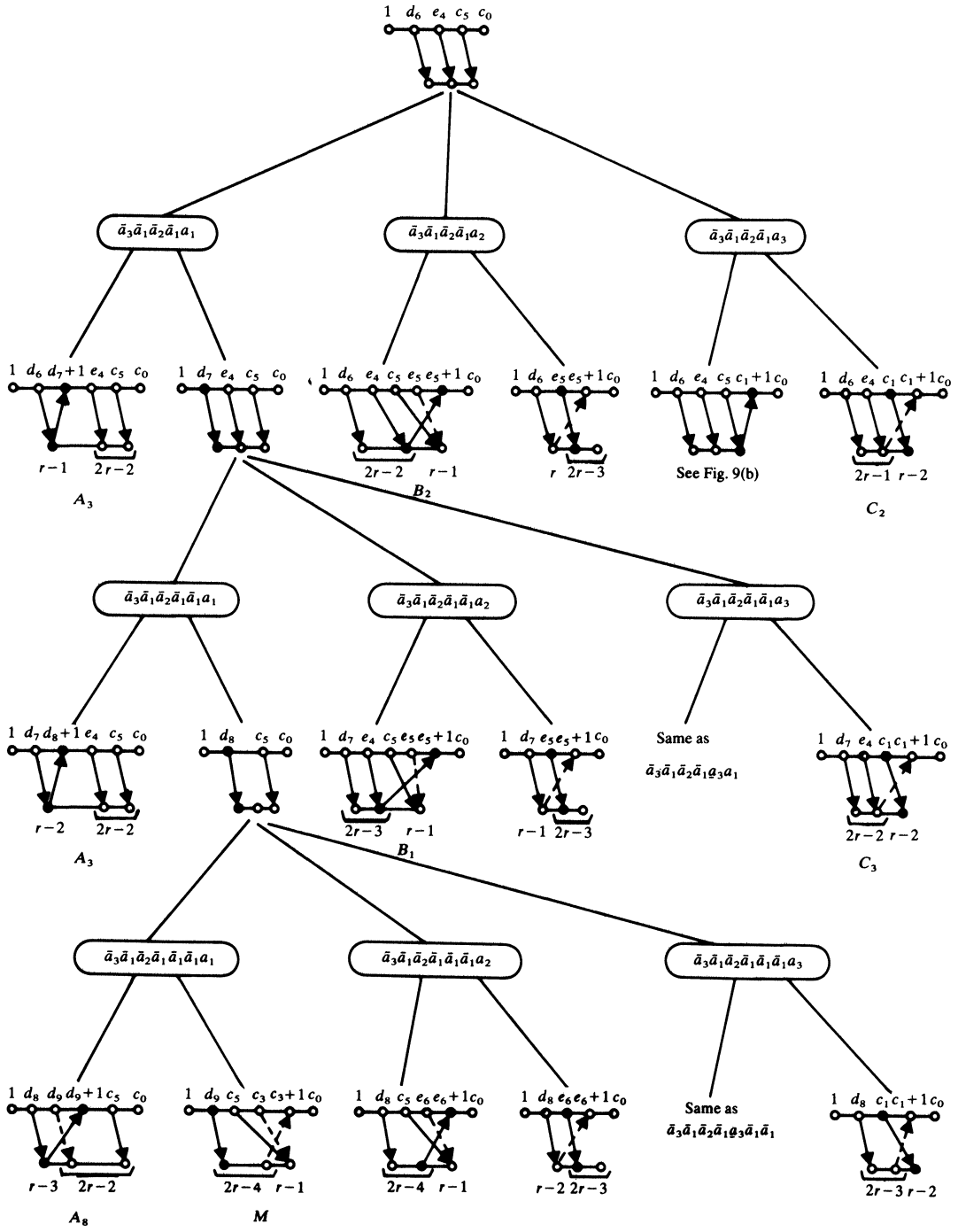
(a)

FIG. 9. *Proof for* $k = 3r+1 \geqq 10$.



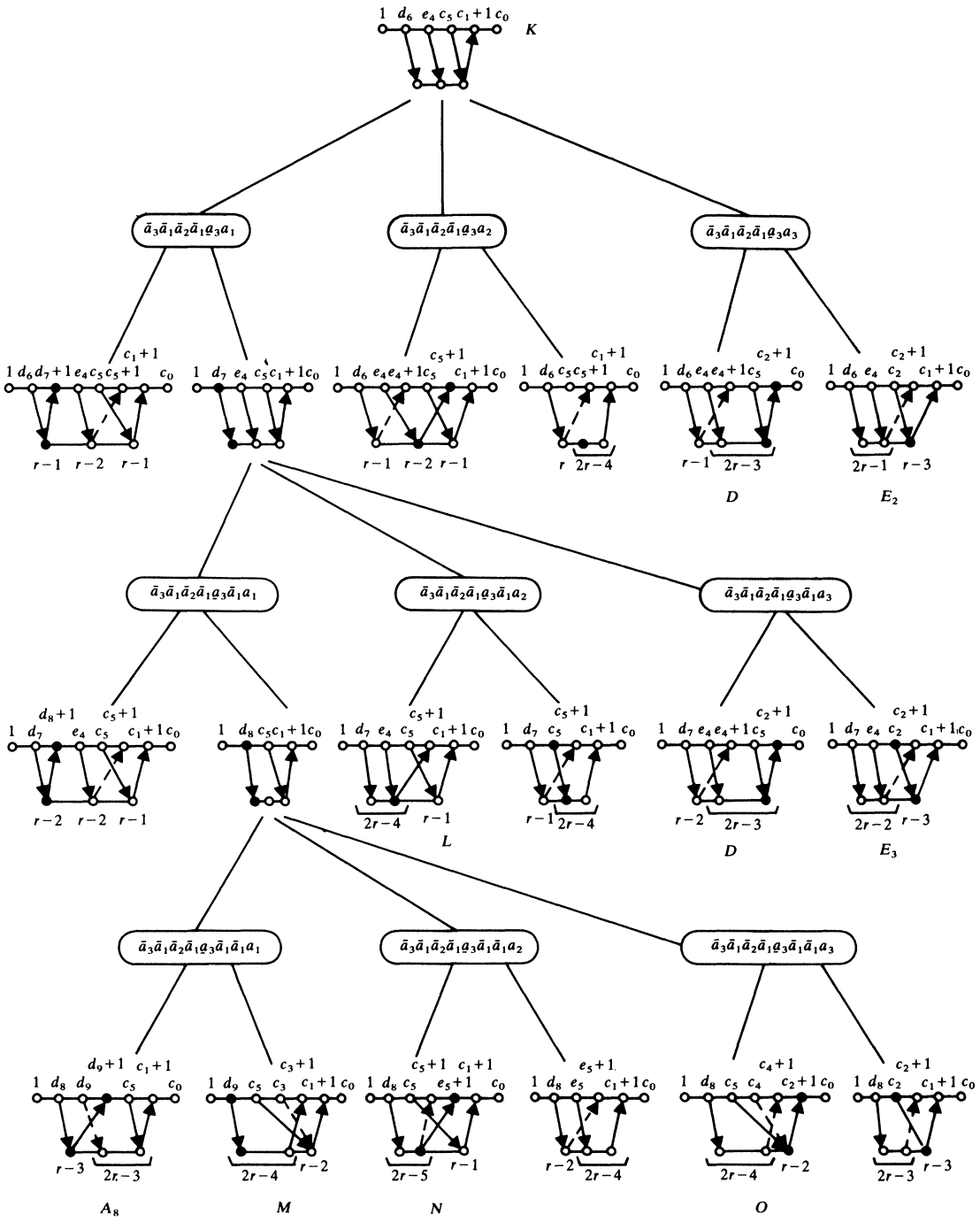
(b)

FIG. 9. *Proof for* $k = 3r+1 \geqq 10$.



(c)

FIG. 9. *Proof for* $k = 3r+1 \geqq 10$.



(d)

FIG. 9. *Proof for* $k = 3r+1 \geqq 10$.

Analysis for labeled nodes in Fig. 9.

A. $c_0 - c_5 = 2^{r-2} + 2^{r-3}$

—A_7. $c_0 - d_4 = [\frac{267}{7} 2^{r-5}] - 1 \geqq [\frac{33}{7} 2^{r-2}] = g(2r-1, r-1, r-3)$ for $r \geqq 4$.

(Configuration A_7 does not occur when $r = 3$, see G.)

—A_6.

$$c_0 - d_3 = \begin{cases} 9 & (\text{for } r = 3) \\ [\frac{137}{7} 2^{r-4}] - 1 & (\text{for } r \geqq 4) \end{cases}$$

$$\geqq [\tfrac{33}{7} 2^{r-2}] = g(2r-1, r-1, r-3).$$

—A_4. $c_0 - d_2 = [\frac{72}{7} 2^{r-3}] - 1 \geqq [\frac{33}{7} 2^{r-2}] = g(2r-1, r-1, r-3)$.

—A_1. $c_0 - d_1 = c_0 - d_2 \geqq g(2r-1, r-1, r-3)$.

—A_5. $c_0 - e_3 = [\frac{19}{7} 2^{r-2}] = g(2r-2, r-1, r-3)$.

—A_2. $c_0 - e_2 \geqq c_0 - e_3 = g(2r-2, r-1, r-3)$.

—A_8.

$$c_0 - d_9 = \begin{cases} 3 & (\textit{for } r = 3) \\ [\frac{53}{7} 2^{r-4}] - 1 & (\textit{for } r \geqq 4) \end{cases}$$

$$\geqq [\tfrac{12}{7} 2^{r-2}] = g(2r-3, r-1, r-3).$$

—A_3. $c_0 - e_4 = [\frac{31}{7} 2^{r-3}] \geqq [\frac{12}{7} 2^{r-2}] = g(2r-3, r-1, r-3)$.

B. $e_5 - e_4 \geqq [\frac{3}{7} 2^{r-1}] \geqq 2^{r-3} + 2^{r-4}$.

—B_1. $e_5 - d_1 \geqq \left[\dfrac{15 \cdot 2^{r-1} - 2}{7}\right] \geqq [\frac{59}{7} 2^{r-3}] = g(2r-2, r-2, r-4)$.

—B_2. $e_5 - d_6 = e_5 - d_1 - 2^{r-1} \geqq g(2r-2, r-2, r-4) - 2^{r-1} = g(2r-3, r-2, r-4)$.

—B_3. $e_5 - d_7 = e_5 - d_6 - 2^{r-2} \geqq g(2r-3, r-2, r-4) - 2^{r-2} = g(2r-4, r-2, r-4)$.

C. $c_1 - e_4 = [\frac{24}{7} 2^{r-3}] \geqq 2^{r-2} + 2^{r-3}$.

—C_1. $c_1 - d_1 = [\frac{72}{7} 2^{r-3}] - 1 \geqq [\frac{33}{7} 2^{r-2}] = g(2r-1, r-1, r-3)$.

—C_2. $c_1 - d_6 = c_1 - d_1 - 2^{r-1} \geqq g(2r-1, r-1, r-3) - 2^{r-1} = g(2r-2, r-1, r-3)$.

—C_3. $c_1 - d_7 = c_1 - d_6 - 2^{r-2} \geqq g(2r-2, r-1, r-3) - 2^{r-2} = g(2r-3, r-1, r-3)$.

—C_4. $c_1 - d_8 = c_1 - d_7 - 2^{r-3} \geqq g(2r-3, r-1, r-3) - 2^{r-3} = g(2r-4, r-1, r-3)$.

D. For $r = 3$, $c_2 = c_1$. Therefore it suffices to consider the case $a_3 \leftarrow c_2$ only. Namely, configuration D does not occur when $r = 3$. For $r \geqq 4$,

$$c_2 - c_5 = 2^{r-3} + 2^{r-4}.$$

$$c_2 - e_4 = [\tfrac{41}{7} 2^{r-4}] \geqq [\tfrac{33}{7} 2^{r-4}] = g(2r-4, r-2, r-4).$$

E. For $r = 3$, E_2 and E_3 are covered by K. For E_1, $N(c_2 - e_4, c_2 - d_1) = N(3, 9) \geqq N(2, 9) = N(2, 2 + 2^3 - 1) > 5$ from Lemma 2. The proof for $r = 4$ is illustrated in Fig. 10.

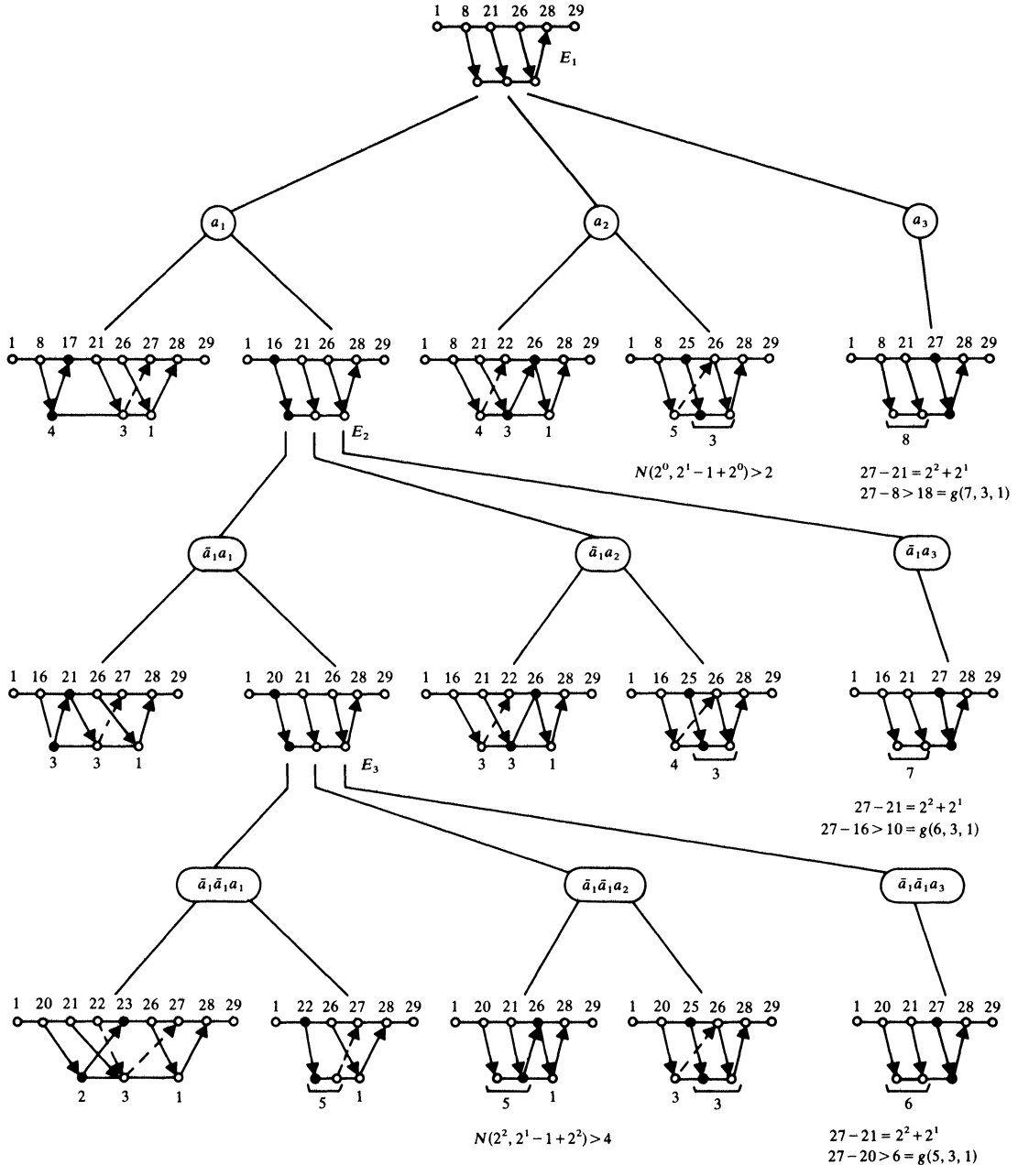


FIG. 10. *Proof for E with* $k = 13$.

For $r \geqq 5$,

$$c_2 - c_4 = [\tfrac{41}{7}2^{r-4}] \geqq 2^{r-2} + 2^{r-4}.$$

—E_1. $c_2 - d_1 = [\tfrac{137}{7}2^{r-4}] - 1 \geqq [\tfrac{17}{7}2^{r-1}] = g(2r-1, r-1, r-4)$.

—E_2. $c_2 - d_6 = c_2 - d_1 - 2^{r-1} \geqq g(2r-1, r-1, r-4) - 2^{r-1} = g(2r-2, r-1, r-4)$.

—E_3. $c_2 - d_7 = c_2 - d_6 - 2^{r-2} \geqq g(2r-2, r-1, r-4) - 2^{r-2} = g(2r-3, r-1, r-4)$.

F. For $r = 3$ and 4, $d_5 - d_1 = 2^{r-2}$.

—F_1. $e_3 - d_1 = 2^{r-1} + 2^{r-2} - 1$. Therefore $N(d_5 - d_1, e_3 - d_1) > (r-1) + (r-2) + 1 = 2r-2$ by Lemma 2.

—F_2. $c_1 - d_1 = 2^r + 2^{r-2} - 1$. Therefore $N(d_5 - d_1, c_1 - d_1) > r + (r-2) + 1 = 2r-1$ by Lemma 2.

For $r \geqq 5$,

$$d_5 - d_1 = [\tfrac{17}{7}2^{r-3}] \geqq 2^{r-2} + 2^{r-5}.$$

—F_1. $e_3 - d_1 = [\tfrac{41}{7}2^{r-3}] = g(2r-2, r-1, r-5)$.

—F_2. $c_1 - d_1 = [\tfrac{72}{7}2^{r-3}] - 1 \geqq [\tfrac{69}{7}2^{r-3}] = g(2r-1, r-1, r-5)$.

G. For $r = 3$, $d_3 = d_2$. Therefore it suffices to consider the case $a_1 \to d_3 + 1$ only.

H. For $r = 3$ and 4, the analysis is similar to F. For $r \geqq 5$,

$$d_5 - d_2 = d_5 - d_1 - 2^{r-3} \geqq 2^{r-3} + 2^{r-5}.$$

—H_1. $e_3 - d_2 = e_3 - d_1 - 2^{r-3} \geqq g(2r-2, r-1, r-5) - 2^{r-3} = g(2r-3, r-2, r-5)$.

—H_2. $c_1 - d_2 = c_1 - d_1 - 2^{r-3} \geqq g(2r-1, r-1, r-5) - 2^{r-3} = g(2r-2, r-2, r-5)$.

I. For $r = 4$, $d_4 = d_3$. Therefore it suffices to consider the case $a_1 \to d_4 + 1$ only.

J. For $r = 4$, the analysis is similar to F. For $r \geqq 5$,

$$d_5 - d_3 = d_5 - d_2 - 2^{r-4} \geqq 2^{r-4} + 2^{r-5}.$$

—J_1. $e_3 - d_3 = e_3 - d_2 - 2^{r-4} \geqq g(2r-3, r-2, r-5) - 2^{r-4} = g(2r-4, r-3, r-5)$.

—J_2. $c_1 - d_3 = c_1 - d_2 - 2^{r-4} \geqq g(2r-2, r-2, r-5) - 2^{r-4} = g(2r-3, r-3, r-5)$.

K. For $r = 3$, the configuration is the following one:

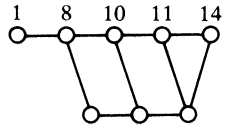


FIG. 10′. *Configuration K for* $r = 3$.

Add a broken link $a_1 \cdots > b_{11}$, then it is easy to check that a_1 needs two comparisons to merge while a_2 and a_3 together needs four. Since five comparisons have already been used, the sum is $11 > 3r + 1 = 10$.

K′. For $r = 4$, the configuration is the following one:

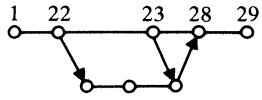


FIG. 11. *Configuration K′ for* $r = 4$.

Seven comparisons have been made and the number of possible orderings of the 29 elements is reduced to 55. However, any further comparison, with one exception, will partition the number of 55 into two numbers with one greater than 32, hence at least six more comparisons, the only exception is when we compare a_2 with b_{25}. Consider the case $b_{25} \to a_2$. Then the new configuration is the following one:

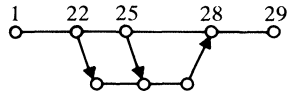


FIG. 12. *A configuration.*

However, when we do the (3, 5) problem and compare a_2 with b_3, regardless of which one is greater, we obtain a configuration equivalent to the above one. Since we know that the (3, 5) problem needs seven comparisons, the above configuration needs six more comparisons.

L. For $r \geqq 4$.

$$c_5 - e_4 = [\tfrac{5}{7}2^{r-2}] \geqq 2^{r-3} + 2^{r-5},$$

$$c_5 - d_7 = [\tfrac{8}{7}2^{r-2}] - 1 \geqq [\tfrac{13}{7}2^{r-3}] = g(2r-5, r-2, r-5).$$

M. For $r \geqq 5$,

$$c_3 - d_9 = [\tfrac{25}{7}2^{r-4}] - 1 \geqq [\tfrac{12}{7}2^{r-3}] = f_2(2r-5) + 1.$$

For $r = 4$, $d_9 = c_5 = 23$. Therefore configuration M without the broken link is the following one:

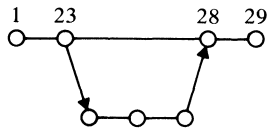


FIG. 13. *Configuration M for* $r = 4$.

which needs six comparisons since it is equivalent to the (3, 4) problem.

M′. Similar to M for $r \geqq 5$.

N. For $r \geqq 5$,

$$c_5 - d_5 = [\tfrac{9}{7}2^{r-3}] - 1 \geqq [\tfrac{17}{7}2^{r-4}] = f_2(2r-6) + 1.$$

O. For $r \geqq 5$,

$$c_4 - d_8 = [\tfrac{25}{7}2^{r-4}] - 1 \geqq [\tfrac{12}{7}2^{r-3}] = f_2(2r-5) + 1.$$

P. If configuration P can be done in less than $2r-3$ comparisons, then we show that configuration A_8 can be done in less than $2r-2$ comparisons, a contradiction to our analysis of A_8. For configuration A_8, we first compare a_3 with $c_1 + 1$. If $a_3 \to c_1 + 1$, then we obtain configuration P which, by assumption, needs less than $2r-3$ configurations. If $c_1 + 1 \to a_3$, then since

$$c_0 - c_1 - 1 = f_1(r-3), \qquad c_0 - d_1 \leqq f_1(r-1),$$

a_3 can be merged in $r-3$ and a_2 in $r-1$ comparisons. In any case, configuration A_8 needs at most $2r-3$ comparisons.

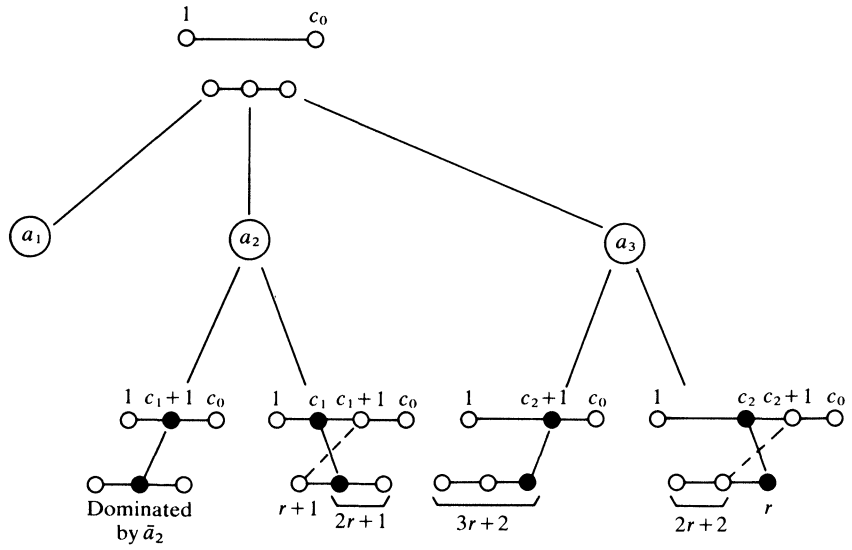


FIG. 14. *Proof for* $k = 3r+2 \geqq 8$.

Finally, we study the case $k = 3r+2 \geqq 8$. The proof is given in Fig. 14. Definitions of symbols used in Fig. 14:

$$c_0 = f_3(3r+2)+1 = \begin{cases} 9 & \text{for } r = 2, \\ \left[\dfrac{17 \cdot 2^r - 6}{7}\right] & \text{for } r \geqq 3, \end{cases}$$

$$c_1 = \left[\frac{c_0}{2}\right] = \begin{cases} 5 & \text{for } r = 2, \\ [\frac{17}{7}2^{r-1}] & \text{for } r \geqq 3, \end{cases}$$

$$c_2 = f_3(3r+1)+1 = [\tfrac{107}{7}2^{r-3}]-1.$$

Analysis for the labeled terminal nodes are straightforward and omitted.

4. Some concluding remarks. Manacher [4], [5] recently gives an algorithm to sort n elements which beats the Ford and Johnson sorting algorithm over infinite many values of n. The Manacher sorting algorithm makes use of optimal merging results, typically when one subset to be merged is very small. Therefore, the knowledge of $f_m(k)$ for small m is likely to help making further improvement over the Manacher sorting algorithm.

With the formulas of $f_3(k)$ now available, we can venture to make some conjecture about the general $f_m(k)$. Noting that

$$\frac{f_1(r)}{f_1(r-1)} \to 2,$$

$$\frac{f_2(2r)}{f_2(2r-1)} \to \frac{17}{12} \sim 1.42, \qquad \frac{f_2(2r+1)}{f_2(2r)} \to \frac{24}{17} \sim 1.41,$$

$$\frac{f_3(3r)}{f_3(3r-1)} \to \frac{43}{34} \sim 1.26 \frac{f_3(3r+1)}{f_3(3r)} \to \frac{107}{86} \sim 1.24, \qquad \frac{f_3(3r+2)}{f_3(3r+1)} \to \frac{136}{107} \sim 1.27,$$

we conjecture

$$\frac{f_m(k)}{f_m(k-1)} \to 2^{1/m}.$$

Since we know

$$f_m(2m) = m+1$$

using the conjecture, we obtain

$$f_m(k) \to (m+1)2^{k/m-2}.$$

Checking with known results for $m = 1, 2, 3$, we note

$$\frac{f_1(k)}{2 \cdot 2^{k-1}} \to 2, \qquad \frac{f_2(k)}{3 \cdot 2^{k/3-2}} \to 1.6, \qquad \frac{f_3(k)}{4 \cdot 2^{k/4-2}} \to 1.5.$$

Therefore the statistic $n \cdot 2^{(k+1)/m-2}$ is not a particularly good estimate of $f_m(k)$.

On the other hand, we note that the numbers $f_2(2r+2)$ differ very little from the numbers $f_3(3r+2)$. In fact,

$$1 \geqq f_2(2r+2) - f_3(3r+2) \geqq 0.$$

Furthermore, since $f_4(6) = 3 = f_3(5)$, $f_4(10) = 8 = f_3(8)$, and the strongly suspected relation $f_m(mr+i) \sim 2f_m(m(r-1)+i)$, we anticipate the numbers $f_4(4r+2)$ will look very much like the numbers $f_3(3r+2)$ and $f_2(2r+2)$. The same can be said for the numbers $f_5(5r+2)$. Although we cannot conclude anything in general, it is possible that a set of numbers similar to $f_2(2r+2)$ will keep turning up in $f_m(k)$ for every m. If that turns out to be true, then using this set of numbers as base plus using the conjecture $f_m(k)/f_m(k-1) \sim 2^{1/m}$, we can come up with a very good estimate for the general $f_m(k)$. Another interesting relation is observed from the following facts:

$$f_1(r+1) - f_1(r) = 2^r, \qquad f_2(2r+1) - f_2(2r) = 2^{r-1}, \qquad f_3(3r+1) - f_3(3r) = 3 \cdot 2^{r-3}.$$

Whether this neat expression exists for $f_m(mr+1) - f_m(mr)$ for all m, or it is purely a coincidental event for these particular values of m is too early to conclude.

REFERENCES

[1] R. L. GRAHAM, *On sorting by comparisons*, Proc. of Atlas Symposium, 2 (1971), pp. 263–269.

[2] F. K. HWANG AND S. LIN, *An optimal algorithm for merging an ordered set of length two with another ordered set*, Acta Information, 1, 2 (1971), pp. 145–158.

[3] D. E. KNUTH, *The Art of Computer Programming*, vol. 3, *Sorting and Searching*, Addison-Wesley, Reading, MA, 1973.

[4] G. K. MANACHER, *The Ford–Johnson sorting algorithm is not optimal*, to appear.

[5] ———, *Nontrivial improvements to the Hwang–Lin algorithm*, to appear.